

Niveau de Distribution		
Limité		2
Interne		3
Externe	X	4

Annuaire Interne des Personnels v 1.0

Manuel d'Installation

(v1.6)

Rédaction		Compléments / Vérification		Approbation		Diffusion	
Nom :	Olivier CHATOR	Nom :		Nom :		Nom :	
Date :	04/01/2010	Date :		Date :		Date :	
Visa :	OC	Visa :		Visa :		Visa :	

1. Historique et documents de référence

1.1. Historique

Date	Version	Nom	Description
25/02/09	S0F0	Olivier CHATOR	Création du document
09/03/2009	1.1	Olivier CHATOR	Intégration des envois de fiches agent modifiées selon le paramétrage défini par la balise « mailMappingChampsModifies »
13/03/2009	1.2	Olivier CHATOR	Ajout étape paramétrage Open LDAP pour AIP_CG33 (page 9, étape 3)
23/04/2009	1.3	Olivier CHATOR	Modification slapd.conf + installation structure d'annuaire vide
06/05/2009	1.4	Olivier CHATOR	Mise à jour suite à : • Changement de JDK (1.4.2 vers 1.5) • Evolution des copies d'écran • Démarrage sans les photos agent en Juin 2009
05/06/2009	1.5	Olivier CHATOR	Intégration doc d'installation de nos collègues des Infrastructures (L. CHELET)
04/01/10	1.6	Olivier CHATOR	Refonte globale

1.2. Documents de référence

Document	Nom
Manuel d'exploitation du logiciel A.I.P v 1.0	AIP v 1.0 - Manuel Exploitation (version).doc
Manuel technique du logiciel A.I.P v 1.0	AIP v 1.0 - Manuel Technique (version).doc
Manuel d'administration du logiciel A.I.P v 1.0	AIP v 1.0 - Manuel Administration (version).doc
Manuel de l'utilisateur du logiciel A.I.P v 1.0	AIP v 1.0 - Manuel Utilisateur (version).doc

Table de matières

1. HISTORIQUE ET DOCUMENTS DE REFERENCE	2
1.1. Historique	2
1.2. Documents de référence	2
2. INTRODUCTION	5
2.1. Objectifs du document	5
3. PHASE 1 – INFRASTRUCTURE TECHNIQUE AU CG33	6
3.1. Configuration matérielle	6
3.2. Initialisation de l'environnement système	6
3.3. Normalisation des installations de logiciels	6
3.4. Utilisation du logiciel Apache 2	7
4. PHASE 2 – INSTALLATION DES LOGICIELS	9
4.1. Introduction	9
4.2. Les produits logiciels indispensables	9
4.3. Localisation des packages d'installation	9
4.4. Pré requis 1 - Installation du JDK	10
4.4.1. Extraction à partir du package d'installation	10
4.4.2. Partie spécifique AIP	10
4.5. Pré requis 2 - Installation Tomcat	11
4.5.1. Extraction à partir du package d'installation	11
4.5.2. Partie spécifique AIP	11
4.5.3. Paramétrage technique	11
4.5.4. Vérification du lancement correct	12
4.6. Pré requis 3 - Installation BerkeleyDB	13
4.6.1. Introduction	13
4.6.2. Extraction à partir du package d'installation	13
4.7. Pré requis 4 - Installation OpenLDAP	13
4.7.1. Extraction à partir du package d'installation	13
4.7.2. Paramétrage technique	14
4.7.3. Spécification du niveau de log OpenLDAP	17
4.7.4. Partie spécifique AIP	18
4.7.5. Vérification du lancement correct	18
4.8. Installation AIP V 1.0	19

4.9.	Paramétrage technique	19
4.10.	Vérification de déploiement correct	20
5.	PHASE 3 - PARAMETRAGE DE BASE OBLIGATOIRE	23
5.1.	Etape N°1 : Arrêtez tomcat	23
5.2.	Etape N°2 : Paramétrage minimal	23
5.3.	Etape N°3 : Effacement manuel de fichiers	24
5.4.	Etape N°4 : Créer le lien vers le montage réseau	24
5.5.	Etape N°5 : Relancez tomcat	24
6.	PHASE 4 - VERIFICATIONS POST-INSTALLATION	25
6.1.	Via la console Manager de Tomcat	25
6.2.	Via l'arborescence de fichier Tomcat	25
6.3.	Via les fichiers de log Tomcat	26
6.4.	Via l'application AIP_CG33 elle-même	27
7.	PHASE 5 - DESINSTALLATION	28
7.1.	Désinstallation	28
7.1.1.	Des pré-requis	28
7.1.2.	De l'AIP via la console d'administration Tomcat	28
7.1.3.	De l'AIP de manière manuelle	29
8.	PHASE 6 – VERIFICATIONS POST DESINSTALLATION	30
8.1.	Via la Console Manager de Tomcat	30
8.2.	Via les Fichiers de log Tomcat	30
9.	ANNEXE : INTERVENTION SUR LES DONNEES DE L'ANNUAIRE	31
9.1.	Introduction	31
9.2.	L'outil JExplorer	31
9.3.	Connexion au Ldap	31
9.4.	Modification de données	32
9.5.	Déconnexion	32

2. Introduction

2.1. Objectifs du document

L'Annuaire Interne des Personnels (désigné par l'acronyme A.I.P.) s'inscrit dans le cadre du projet global « Annuaire du CG33 » qui regroupe également les annuaires applicatifs internes et externes (SSO).

Dans ce contexte, le présent document vise à décrire tous les aspects techniques relatifs à l'installation et la désinstallation du logiciel A.I.P. en v 1.0. ainsi qu'à la description des méthodes de vérification que ces événements se sont bien déroulés.

3. Phase 1 – Infrastructure technique au CG33

3.1. Configuration matérielle

Composants	Produit
Infrastructure	Serveur : serveur IP : 0.0.0.0 Type : machine Virtuelle VMware sur le cluster VMWARE CG33 Caractéristiques : 1 processeur 2,66GHz/ 4 Mo de cache 4 Go de RAM
Système d'exploitation	RedHat Entreprise 5.1 GNU/Linux SMP x86_64 2.6.18-128.1.1.el5
Serveur http	Apache 2.2.11 / Tomcat 6.0.20
Serveur d'applications	Tomcat 6.0.20 / Openldap 2.0.27, BerkeleyDatabase 4.2.52
J2EE : Plate-forme Java	jdk 1.6. 0_16 (Compilé AMD 64 bits)

3.2. Initialisation de l'environnement système

1) Création d'un serveur de type machine virtuelle VMware dédié depuis VCenter
(Modèle : **RHEL 5.x 64 bits**).

2) Création de groupes **tomcat**, **appli** et **openldap**, ce qui donne dans le **/etc/group** :

```
tomcat:x:501:aipl,expl
appli:x:1000:tomcat,daemon,nagios,expl
openldap:x:502:aipl
```

3) Création des utilisateurs **aipl**, **tomcat** et **openldap**, ce qui donne dans le **/etc/passwd** :

```
tomcat:x:501:501:Serveur Tomcat:/opt/tomcat:/bin/bash
openldap:x:502:502:Serveur OpenLdap:/opt/openldap:/bin/bash
aipl:x:1001:1000:/home/aipl:/bin/bash
```

3.3. Normalisation des installations de logiciels

Les logiciels sont installés dans **/opt**. Pour chacun d'eux, des liens symboliques sont créés.

```
lrwxrwxrwx 1 root root 12 mai 28 2009 apache2 -> httpd-2.2.11
drwxr-xr-x 9 tomcat tomcat 4096 oct 30 11:03 apache-tomcat-6.0.20
lrwxrwxrwx 1 root root 9 mai 28 2009 berkeleydb -> db-4.2.52
drwxr-xr-x 6 root root 4096 mai 28 2009 db-4.2.52
drwxr-xr-x 14 root root 4096 mar 12 2009 httpd-2.2.11
lrwxrwxrwx 1 root root 11 d'Ãc 14 13:27 jdk -> jdk1.6.0_16
drwxr-xr-x 10 root root 4096 sep 23 15:06 jdk1.6.0_16
lrwxrwxrwx 1 openldap openldap 15 mai 28 2009 openldap -> openldap-2.0.27
drwxr-xr-x 11 openldap openldap 4096 aoÃ 21 10:09 openldap-2.0.27
lrwxrwxrwx 1 root root 22 oct 30 11:33 tomcat -> ./apache-tomcat-6.0.20
```

A Noter :

- Le JDK est téléchargé en version binaire « installeable » depuis le site de Sun, en version 64 bits optimisé pour processeur AMD. En effet, AMD est le processeur identifié sur les machines virtuelles VM WARE
- Le logiciel TOMCAT est à la dernière révision de la version 6.0. La version 6.0.20 est validée pour les environnements de production. Elle fonctionne avec Java 1.5 ou supérieur.

3.4. Utilisation du logiciel Apache 2

Le logiciel APACHE 2 est compilé avec les options suivantes :

```
# root
Prérequis : Apr.1.3.3
# ./configure && make && make install

Prérequis : Aprutil.1.3.4
# ./configure --with-apr=/usr/local/apr --with-openssl --with-ldap && make && make install

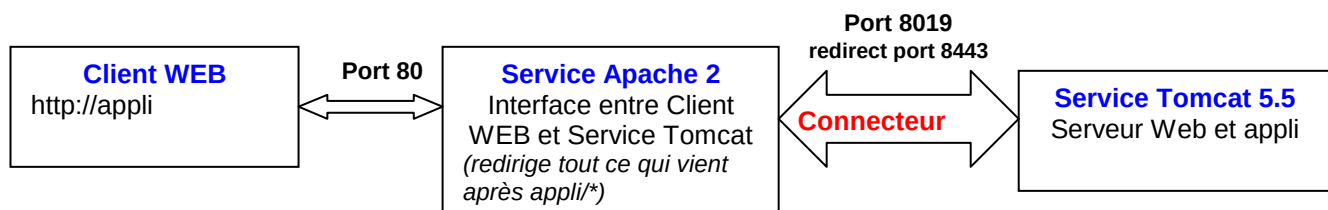
# ./configure --prefix=/opt/httpd-2.2.11 --enable-mods-shared="all ssl ldap info headers proxy
proxy-balancer proxy-ajp rewrite authnldap cache mem-cache file-cache disk-cache log-agent expires"

# make

# make install
```

Les modules non utilisés seront désactivés dans le fichier **/opt/apache2/conf/httpd.conf**

On procède à l'installation du connecteur AJP **tomcat-connector-1.2.27** (compilé à partir des sources). Le connecteur AJP (ou mod_jk) permet la redirection des flux http et https entre le serveur apache et le serveur tomcat comme indiqué dans le schéma suivant :



```
# (root)
# cd ./native

# ./configure --prefix=/opt/httpd-2.2.11/tomcat-connector-1.2.27 --with-apxs=/opt/httpd-
2.2.11/bin/apxs --with-gnu-ld --with-java-home=/opt/jdk
--enable-shared --enable-EAPI

# make

# libtool --mode=finish /opt/httpd-2.1.11/modules
```

Modification du fichier **/opt/apache2/conf/httpd.conf** pour prendre en compte le module mod_jk (le connecteur).

```
# httpd.conf
listen 80

loadModule jk_module modules/mod_jk.so

<IfModule jk_module>
include conf/mod_jk.conf
</IfModule>
```

Ajout du fichier **/opt/apache2/conf/mod_jk.conf**
(user et group tomcat :tomcat)

```
# mod_jk.conf
# Where to find workers.properties
JkWorkersFile conf/workers.properties
```

```
# Send SSL key
# JkOptions +ForwardKeySize +ForwardURISCompat -ForwardDirectories

# Where to put jk shared memory
JkShmFile      logs/mod_jk.shm
# Size Mb
JkShmSize 1

# Where to put jk logs
JkLogFile      logs/mod_jk.log

# Set the jk log level [debug/error/info]
JkLogLevel     info

# Select the timestamp log format
JkLogStampFormat "[%a %b %d %H:%M:%S %Y]"
JkRequestLogFormat "%w %V %T"
```

Ajout du fichier /opt/apache2/conf/worker.properties :
(user et group tomcat :tomcat)

```
# worker.properties
# Define 1 real worker using ajp13
worker.list=worker1
# Set properties for worker1
worker.worker1.type=ajp13
worker.worker1.host=localhost
worker.worker1.port=8019
```

Le paramétrage technique de Tomcat qui lui permet d'utiliser modjk est présenté au chapitre suivant.

4. Phase 2 – Installation des logiciels

4.1. Introduction

Les opérations d'installation décrites ici sont destinées à un serveur Linux. Pour un serveur Windows, chaque logiciel de type « pré-requis » possède son propre installateur automatique que l'on utilisera.

4.2. Les produits logiciels indispensables

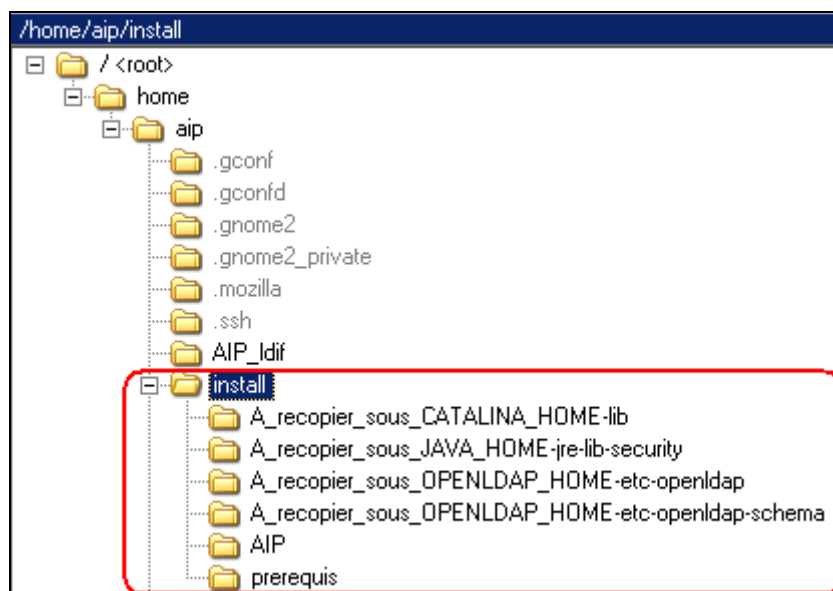
Pour pouvoir être exécuté de manière satisfaisante, le logiciel A.I.P. v 1.0 nécessite de disposer, sur le serveur (**machine virtuelle Linux** par exemple avec **4Go** de RAM si possible) qui l'héberge :

- 1) D'un serveur d'application web de type Tomcat en version 6.0.20 à minima (racine d'installation : **CATALINA_HOME**)
- 2) D'un **JDK 1.6** (les composants logiciels ayant été compilés avec la version 1.6.0-16)
- 3) D'un accès à l'annuaire LDAP qui héberge les données de l'annuaire lui-même. On parle bien ici d'un accès, car il n'est pas obligatoire que le serveur LDAP soit lui-même installé sur le même serveur à partir du moment où le LDAP est accessible via une url de la forme **ldap://serveur :port**. A titre d'illustration, nous avons réalisé les développements sur un serveur **OpenLDAP 2.0.27**

On considère dans tout ce qui suit que tous ces pré-requis sont remplis

4.3. Localisation des packages d'installation

Tous les packages nécessaires à l'installation (**hors JDK 1.6.0-16 que l'on considère comme aisément accessible sur le site de Sun** en cliquant [ici](#)) sont disponibles sur le serveur linux de VA (**serveur** au CG33) sous le répertoire **/home/aip/install** :



On notera que les sous-répertoires de **/home/aip/install** contiennent :

Répertoire	Contenu	Commentaire
AIP	AIP_CG33.tar.gz axis.tar.gz	Webapp de l'IHM de consultation AIP Webapp des Web Services AIP
prerequis	apache-tomcat-6.0.20.tar.gz db-4.2.52.tar.gz openldap-2.0.27.tar.gz	Tomcat 6.0.20 BerkeleyDB (utilisé par OpenLDAP) OpenLDAP 2.0.27
A_recopier_sous_CATALINA_HOME-lib	activation-1.1.jar commons-dbc-1.2.2.jar jsp-api.jar ojdbc5.jar xercesImpl.jar commons-collections-3.2.1-bin.jar commons-pool-1.5.2-bin.jar mail.jar servlet-api.jar xml-apis.jar	packages java à recopier sous \${CATALINA_HOME}/lib une fois que Tomcat aura été installé
A_recopier_sous_JAVA_HOME-jre-lib-security	local_policy.jar US_export_policy.jar	Packages java de cryptographie à recopier sous \${JAVA_HOME}/jre/lib/security une fois le JDK installé
A_recopier_sous_OPENLDAP_HOME-etc-openldap	slapd.conf	Fichier à recopier sous \${LDAP_HOME}/etc/openldap une fois OpenLDAP installé
A_recopier_sous_OPENLDAP_HOME-etc-openldap-schema	inetorgperson.schema	Fichier à recopier sous \${LDAP_HOME}/etc/openldap/schema une fois OpenLDAP installé

4.4. Pré requis 1 - Installation du JDK

4.4.1. Extraction à partir du package d'installation

On utilise ici la procédure standard définie en lien avec les collègues des infrastructures où **\${JAVA_HOME}** pointe vers le lien symbolique **/opt/jdk**. Il n'y a pas de particularités à signaler pour cette étape d'installation standard d'un JDK.

Vérification file system :

La distribution du jdk doit être présente sous **\${JAVA_HOME}**

4.4.2. Partie spécifique AIP

Dans le cadre pur de l'AIP, on peut dès à présent procéder à la recopie des fichiers de la distribution d'installation, répertoire **A_recopier_sous_JAVA_HOME-jre-lib-security**, vers le répertoire cible du JDK **\${JAVA_HOME}/jre/lib/security**

4.5. Pré requis 2 - Installation Tomcat

4.5.1. Extraction à partir du package d'installation

Enchaînez les opérations suivantes une fois connecté avec l'utilisateur **tomcat** (groupe **tomcat**) :

- 1) Se placer sous **/opt**
- 2) Lancez la commande **cp /home/aip/install/prerequis/apache-tomcat-6.0.20.tar.gz .**
- 3) Lancez la commande **gzip -d apache-tomcat-6.0.20.tar.gz**
- 4) Lancez la commande **tar xvf apache-tomcat-6.0.20.tar**
- 5) Supprimez le fichier **jakarta-tomcat-6.0.20.tar**
- 6) Vérifier (ou créer) le lien **/opt/tomcat** qui pointe vers **/opt/apache-tomcat-6.0.20**

```
drwxr-xr-x  9 tomcat  tomcat  4096 oct 30 11:03 apache-tomcat-6.0.20
lrwxrwxrwx  1 root   root    22 oct 30 11:33 tomcat -> ./apache-tomcat-6.0.20
```

Vérification file system : La distribution Tomcat doit être présente sous **\${CATALINA_HOME}**

4.5.2. Partie spécifique AIP

Dans le cadre pur de l'AIP, on peut dès à présent procéder à la recopie des fichiers de la distribution d'installation, répertoire **A_recopier_sous_CATALINA_HOME-lib**, vers le répertoire cible **\${CATALINA_HOME}/lib**

4.5.3. Paramétrage technique

On enchaîne les opérations suivantes :

- 1) Génération de la clé RSA de l'utilisateur tomcat (nécessaire pour le fonctionnement du connecteur modjk)

```
# su - tomcat

$ keytool -genkey -alias tomcat -keyalg RSA

mdp et keystore : tomcat
nom : tomcat
UO : dsi
Org : cg33
Ville : bordeaux
Prov : gironde
Pays : fr
```

- 2) Modification du fichier **/opt/tomcat/conf/server.xml** pour changer le port TCP pour l'écoute AJP en 8019

```
<!-- Define an AJP 1.3 Connector on port 8009 -->
<Connector port="8019" enableLookups="false" redirectPort="8443" protocol="AJP/1.3" />
```

- 3) Modification du fichier **/opt/tomcat/conf/server.xml** pour « tuner » le serveur tomcat

```
<!-- Define a SSL HTTP/1.1 connector on port 8443 -->
<Connector port="8443"      maxHttpHeaderSize="8192"
  maxThreads="400" minSpareThreads="25" maxSpareThreads="75"
  maxKeepAliveRequests="100" connectionTimeout="20000"
  enableLookups="false" disableUploadTimeout="true"
  clientAuth="false" sslProtocol="TLS" />

<Host name="localhost" appBase="webapps"
  unpackWARs="true" autoDeploy="false"
  xmlValidation="false" xmlNamespaceAware="false">
```

4) Ajout d'un lien symbolique pour la library **/opt/jdk/lib/tools.jar** dans **/opt/tomcat/common/lib/** pour que les compilations dynamiques des modules jsp. *Alternativement on aurait pu ajouter une variable d'environnement CLASSPATH.*

```
# su - tomcat

$ cd /opt/tomcat/common/lib/
$ ln -s /opt/jdk/lib/tools.jar tools.jar
```

5) Ajout du fichier **/opt/tomcat/bin/setenv.sh** (user et group **tomcat :tomcat**)
Outre la définition des "HOME" JAVA et TOMCAT, on paramètre les ressources mémoire allouées à JAVA / TOMCAT pour éviter les plantages lors des compilations dynamiques (Exception : No heap memory).

```
# setenv.sh
JAVA_HOME=/opt/jdk
CATALINA_HOME=/opt/tomcat
JAVA_OPTS="-Xms512M -Xmx512M"

Export JAVA_HOME CATALINA_HOME JAVA_OPTS
```

6) Mise en place des scripts de démarrage

- **/etc/init.d/tomcat**
- **/etc/init.d/httpd**

7) Mise en service des démon **tomcat** et **httpd** avec la commande **chkconfig** pour démarrage automatique.

```
# (root)
# chkconfig --add tomcat
# chkconfig --list tomcat
# chkconfig --add httpd
# chkconfig --list httpd
```

4.5.4. Vérification du lancement correct

Suite aux opérations ci-dessus, on procède au démarrage des services **httpd** et **tomcat** (java) avec vérification :

```
# (root)
# service tomcat start
# service tomcat status
  running in normal mode    [ OK ]
  java (pid xxxx) en cours d'exécution...

# ps -ef |grep tomcat
/opt/jdk/bin/java -Djava.util..... -Xms513M -Xmx521M ..... start

# service httpd start
# service httpd status
apache2 status :
  httpd (pid xxxx xxxx xxx xxx ) en cours d'exécution...

# netstat -an | grep LISTEN | grep 80

tcp 0  0  :::80                :::*                LISTEN
tcp 0  0  :::8080              :::*                LISTEN
tcp 0  0  :::8019              :::*                LISTEN
```

Il est également possible de lancer la commande **ps -ef | grep catalina | grep -v grep** pour obtenir une ligne de la forme :

```
oc          24693          1    0 10:18 pts/2          00:00:27 /usr/local/jdk/bin/java -
Djava.endorsed.dirs=/home/oc/tomcat/common/endorsed -classpath
/usr/local/jdk/lib/tools.jar:/home/oc/tomcat/bin/bootstrap.jar:/home/oc/tomcat/bin/com
mons-logging-api.jar -Dcatalina.base=/home/oc/tomcat -Dcatalina.home=/home/oc/tomcat -
Djava.io.tmpdir=/home/oc/tomcat/temp org.apache.catalina.startup.Bootstrap start
```

4.6. Pré requis 3 - Installation BerkeleyDB

4.6.1. Introduction

L'installation de ce logiciel est impérative pour OpenLDAP. On doit donc réaliser 2 étapes d'installation et dans cet ordre :

- Etape 1 : BerkeleyDB
- Etape 2 : OpenLDAP

4.6.2. Extraction à partir du package d'installation

On enchaîne les opérations suivantes en tant que user **root** :

- 1) Créer un répertoire d'installation **/opt/db-4.2.52_install** et vous y placer
- 2) Lancez la commande **cp /home/aip/install/prerequis/db-4.5.20.tar.gz .**
- 3) Lancez la commande **gzip -d db-4.5.20.tar.gz**
- 4) Lancez la commande **tar xvf db-4.5.20.tar** (un répertoire **db-4.2.52** est créé à la racine)
- 5) Supprimez le fichier **/opt/db-4.2.52_install/db-4.5.20.tar**
- 6) Lancez la commande **cd db-4.2.52/build_unix**
- 7) Lancez la commande **./dist/configure --prefix=/opt/db-4.2.52 --enable-java** où l'option **--prefix** désigne le répertoire cible d'installation de BerkeleyDB
- 8) Lancez la commande **make**
- 9) Lancez la commande **make install**
- 10) Supprimer le répertoire **/opt/db-4.2.52_install** et tout ce qu'il contient
- 11) Vérifier (ou créer) le lien **/opt/berkeleydb** qui pointe vers **/opt/db-4.2.52**

```
lrwxrwxrwx 1 root root 9 -- ---- berkeleydb -> db-4.2.52
drwxr-xr-x 14 root root 4096 -- ---- db-4.2.52
```

Vérification file system :

La distribution BerkeleyDB doit être présente sous **\${BERKELEY_HOME}** qui vaut **/opt/berkeleydb**

4.7. Pré requis 4 - Installation OpenLDAP

4.7.1. Extraction à partir du package d'installation

On enchaîne les opérations suivantes en tant que user **root** :

- 1) Placez-vous sous **/opt** et créer un répertoire **/opt/oldap_install**
- 2) Lancez la commande **cd /opt/oldap_install**
- 3) Lancez la commande **cp /home/aip/install/prerequis/openldap-2.0.27.tar .**
- 4) Lancez la commande **tar xvf openldap-2.0.27.tar** (un répertoire **openldap-2.0.27**)
- 5) Supprimer le fichier **/home/oc/openldap-2.0.27.tar**
- 6) Lancez la commande **cd openldap-2.0.27**
- 7) Lancez la commande **export HOME_LDAP=/opt/openldap-2.0.27**
- 8) Lancez la commande **export LDFLAGS=-L/opt/berkeleydb/lib**
- 9) Lancez la commande **export CPPFLAGS=-I/opt/berkeleydb/include**
- 10) Lancez la commande **./configure --prefix=/opt/openldap-2.0.27**

- 11) Lancez la commande **make depend**
- 12) Lancez la commande **make**
- 13) Lancez la commande **make install**
- 14) Supprimer le répertoire **/opt/oldap_install** et tout ce qu'il contient

Vérification file system :

La distribution Open LDAP doit être présente sous **\${LDAP_HOME}** (qui vaut **/opt/openldap**)

4.7.2. Paramétrage technique

On rappelle que l'on créé plus haut un compte et un groupe local sont créés pour le service ldap :

```
# root
# useradd -u 502 -d /opt/openldap openldap

# chown -R openldap:openldap /opt/openldap-2.0.27
```

On procède également à la création du script d'initialisation **/etc/init.d/openldap**

```
#!/bin/sh
#
# Init file for ldap server daemon (slapd and slurpd).
#
# chkconfig: 2345 93 93
# description: LDAP Lightweight Directory Access Protocol Server.
# processname: slapd
# config: /etc/openldap/slapd.conf
# pidfile: /var/run/slapd.pid
# Laurent CHELET CG33/DGAF/DSI/BASIR
# Modified 20090603

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

RETVAL=0
prog="slapd"
owner="openldap"
lockfile="/var/lock/subsys/$prog"
pidfile="/var/run/$prog.pid"
THEPORT=0000
HOME_OPENLDAP="/opt/openldap"
CONFFILE="$HOME_OPENLDAP/etc/openldap/slapd.conf"

start() {
    # Start daemons.
    SHUTDOWN_PORT=`netstat -vatn|grep LISTEN|grep $THEPORT |wc -l`
    if [ $SHUTDOWN_PORT -ne 0 ]; then
        echo "$prog already started"
    else
        echo -n "Starting $prog : "
        echo
        su $owner -c "$HOME_OPENLDAP/libexec/$prog -h ldap://$HOSTNAME:$THEPORT -f $CONFFILE"
        RETVAL=$?
        [ $RETVAL -eq 0 ] && {
            touch $lockfile
            ps -C $prog -o pid= > $pidfile
            success $prog
        }
    fi
}
```

```

        }
        echo -n "$prog stated in normal mode"
    fi
    return $RETVAL
}

stop() {
    # Stop daemons.
    echo -n "Shutting down $prog:"
    killproc $prog
    RETVAL=$?
    [ $RETVAL -eq 0 ] && {
        rm -f $lockfile
        rm -f $pidfile
    }
    return $RETVAL
}

reload() {
    #Reload
    echo -n "Reloading $prog..."
    if [ -e $pidfile ]; then
        kill -HUP `cat $pidfile`
        RETVAL=$?
        return $RETVAL
    else
        echo " failure. $pidfile not present"
    fi
}

etat() {
    # Status
    SHUTDOWN_PORT=`netstat -vatn|grep LISTEN|grep $THEPORT|wc -l`
    if [ $SHUTDOWN_PORT -eq 0 ]; then
        echo -n "$prog stopped"
        echo
    else
        echo -n "$prog is running"
        echo
        status $prog
    fi
    return $RETVAL
}

case "$1" in
start)
    start
    echo
;;
stop)
    stop
    echo
;;
restart)
    stop $$ sleep 3
    echo
    start
    echo
;;
reload)
    reload
    echo
;;
status)
    etat
    echo
;;
*)
    echo "Usage: $0 {start|stop|restart|reload|status}"
    exit 1
esac

```

```
exit $RETVAL
```

On met finalement en service le démon **openldap** avec la commande **chkconfig** pour un démarrage automatique.

```
# (root)
# chkconfig --add openldap
# chkconfig --list openldap
# service openldap start
```

4.7.3. Spécification du niveau de log OpenLDAP

Il est possible d'activer les logs OpenLDAP qui sont désactivés par défaut.

1) Editer le fichier **/opt/openldap/etc/openldap/slapd.conf** et ajouter à la fin :

```
# Niveau de Log openldap
loglevel 256
```

2) Editer le fichier **/etc/syslog.conf** et ajouter à la fin :

```
# Log slapd openldap
local4.* -/var/log/slapd.log
```

3) Relancer syslog :

```
/etc/init.d/syslog restart
```

4) Relancer openldap :

```
/etc/init.d/syslog restart
```

Pour info :

Niveaux de log OpenLDAP :

number	hex-value	log-name	Logging description
-1	0xFFFF		enable all logging
0	0x0000	-	logging inhibited - no logging occurs including critical errors. Not recommended.
1	0x1	acl	trace function calls
2	0x2	packets	debug packet handling
4	0x4	args	heavy trace debugging
8	0x8	conns	connection management
16	0x10	BER	print out packets sent and received
32	0x20	filter	search filter processing
64	0x40	config	configuration file processing
128	0x80	ACL	access control list processing
256	0x100	stats	stats log connections/operations/results
512	0x200	stats2	stats log entries sent
1024	0x400	shell	print communication with shell backends
2048	0x800	parse	print entry parsing debugging
4096	0x1000	cache	caching (unused)
8192	0x2000	index	indexing (unused)
16384	0x4000	sync	print syncrepl (replica) logging
32768	0x8000	none	A misnomer - it will log message that are not categorized including curial messages

On les positionne dans le slapd.conf avec **loglevel v** où **v** peut être la somme de différents niveaux.

Exemple :

```
loglevel 255
# sets 1, 2, 4, 8, 16, 32, 64 and 128
# adds all the numbers
```

2) Par exemple, avec un loglevel **256**, le fichier **/var/log/slapd.log** contient après redémarrage et une recherche :

```
Jan 12 12:05:48 serveur slapd[21601]: slapd shutdown: waiting for 0 threads to terminate
Jan 12 12:05:49 serveur slapd[21601]: slapd stopped.
Jan 12 12:05:50 serveur slapd[13639]: slapd starting
Jan 12 12:08:02 serveur slapd[13639]: daemon: conn=0 fd=7 connection from IP=172.18.2.44:40135
(IP=172.18.2.44:0000) accepted.
Jan 12 12:08:02 serveur slapd[13639]: conn=0 op=0 BIND dn="CN=CG33,DC=CG33,DC=CG33"
method=128
Jan 12 12:08:02 serveur slapd[13639]: conn=0 op=0 RESULT tag=97 err=0 text=
Jan 12 12:08:03 serveur slapd[13639]: conn=0 op=1 SRCH base="dc=cg33,dc=cg33" scope=2
filter="(ou=*)"
Jan 12 12:08:07 serveur slapd[13639]: conn=0 op=1 SEARCH RESULT tag=101 err=0 text=
Jan 12 12:08:15 serveur slapd[13639]: conn=0 op=2 SRCH base="dc=cg33,dc=cg33" scope=2
filter="(sn=chator*)"
Jan 12 12:08:15 serveur slapd[13639]: conn=0 op=2 SEARCH RESULT tag=101 err=0 text=
Jan 12 12:08:18 serveur slapd[13639]: conn=0 op=3 SRCH
base="ou=DGAF_DSI_SPSI,ou=DGAF_DSI,ou=DGAF,dc=cg33,dc=cg33" scope=2
filter="(employeeNumber=9603158)"
Jan 12 12:08:18 serveur slapd[13639]: conn=0 op=3 SEARCH RESULT tag=101 err=0 text=
Jan 12 12:08:18 serveur slapd[13639]: conn=0 op=5 SRCH base="dc=cg33,dc=cg33" scope=2
filter="(manager=employeeNumber=9603158)"
Jan 12 12:08:19 serveur slapd[13639]: conn=0 op=6 SRCH base="dc=cg33,dc=cg33" scope=2
filter="(employeeNumber=311522)"
Jan 12 12:08:19 serveur slapd[13639]: conn=0 op=5 SEARCH RESULT tag=101 err=0 text=
Jan 12 12:08:19 serveur slapd[13639]: conn=0 op=6 SEARCH RESULT tag=101 err=0 text=
```

Ce loglevel ne semble pas pénaliser les perfs, mais attention au volume du fichier de log compte tenu du nombre de requêtes par jour !

4.7.4. Partie spécifique AIP

Dans le cadre pur de l'AIP, on peut dès à présent procéder à la recopie des fichiers de la distribution d'installation :

- **A_recopier_sous_OPENLDAP_HOME-etc-openldap**, vers le répertoire cible **\${LDAP_HOME}/etc/openldap**
- **A_recopier_sous_OPENLDAP_HOME-etc-openldap-schema**, vers le répertoire cible **\${LDAP_HOME}/etc/openldap/schema**

4.7.5. Vérification du lancement correct

Suite aux opérations ci-dessus, on procède au démarrage du service **openldap** via la commande :
/etc/init.d/openldap start

En lançant la commande **ps -ef | grep slapd | grep -v grep** vous devriez obtenir :

```
openldap 17216      1  0   2009 ?        00:41:17 /opt/openldap/libexec/slapd -h
ldap://serveur.cg33.fr:0000 -f /opt/openldap/etc/openldap/slapd.conf
```

4.8. Installation AIP V 1.0

Enchaîner les actions suivantes en tant que user **aip** :

- 1) Créer le répertoire **/home/aip/webapps/AIP_CG33**, que l'on nommera **\${AIP_HOME}**, et vous y placer
- 2) Exécuter : **cp /home/aip/install/AIP/AIP_CG33.tar.gz** .
- 3) Exécuter : **gzip -d AIP_CG33.tar.gz**
- 4) Exécuter : **tar -xvf AIP_CG33.tar**
- 5) Exécuter : **rm AIP_CG33.tar**

En tant que user **root** :

- 1) Exécuter : **/etc/init.d/tomcat stop**
- 2) Exécuter : **rm /opt/tomcat/works/Catalina/localhost/AIP_CG33** (si ceci existe)
- 3) Exécuter : **ln -s /home/aip/webapps/AIP_CG33 /opt/tomcat/webapps/AIP_CG33**

4.9. Paramétrage technique

On enchaîne les opérations suivantes :

- 1) Changement des users et groupes sur le répertoire **/home/aip**

```
# root
# chmod g+x /home/aip
# chmod g+w /home/aip/temp
# chmod g+w /home/aip/logs
```

- 2) Modification de la configuration apache : Déclaration de 2 Vhosts.

```
NameVirtualHost *:80
# Virtual Hosts
<VirtualHost *:80>
DocumentRoot "/opt/apache2/htdocs"
ServerName serveur.cg33.fr
JkMount /AIP_CG33 worker1
JkMount /AIP_CG33/* worker1
</VirtualHost>

# Virtual Host AIP_CG33 intrannuaire
<VirtualHost *:80>
ServerName intrannuaire.cg33.fr
ServerAlias intrannuaire
RewriteEngine on
RewriteRule ^/$ /AIP_CG33/ [R]
JkMount /AIP_CG33 worker1
JkMount /AIP_CG33/* worker1
</VirtualHost>
```

- 3) Relancer le serveur apache

```
# (root)
# service httpd stop
# service httpd start
```

- 4) Pour la production (serveur serveur), créer un alias name dans le DNS interne du CG33. Créer un alias CNAME : **intrannuaire = serveur.cg33.fr**

- 5) Ajout des droits en écriture pour le groupe appli sur **~aip/webapps/AIP_CG33/logs**
(nb: tomcat fait partie de ce groupe appli et a besoin des droits pour écrire et créer les fichiers log de l'appli AIP)

5') **Spécifique CG33** : Ajout des droits en écriture pour le groupe appli sur **~aip/webapps/AIP_CG33/temp**
(nb: tomcat fait partie de ce groupe appli et a besoin des droits pour écrire et créer les fichiers à envoyer à Abyla)

6) **Mise en place de log4j**

* 2 librairies .jar dans **/opt/tomcat/common/lib** :

```
-rw-r--r-- 1 tomcat tomcat 60841 jui 1 16:40 commons-logging-1.1.1.jar
lrwxrwxrwx 1 root root 25 jui 1 16:41 commons-logging.jar -> commons-logging-1.1.1.jar
-rw-r--r-- 1 tomcat tomcat 391834 jui 1 16:39 log4j-1.2.15.jar
lrwxrwxrwx 1 root root 16 jui 1 16:41 log4j.jar -> log4j-1.2.15.jar

lrwxrwxrwx 1 tomcat tomcat 22 mai 28 17:01 tools.jar -> /opt/jdk/lib/tools.jar
(requis pour éviter le plantage possible lors de l'instantiation de la webapps)
```

* Création du fichier log4j.properties dans **opt/tomcat/common/classes**

Détail du fichier :

```
#
# Configuration du Log4j pour le log de Tomcat
# Laurent Chelet CG33/DSI le 01/07/2009
#

# Configure the logger to output info level messages into a rolling log file.
log4j.rootLogger=WARN, R
log4j.rootLogger=INFO, A1

# configuration for standart output (catalina.out)
log4j.appender.A1=org.apache.log4j.ConsoleAppender
log4j.appender.A1.layout=org.apache.log4j.PatternLayout
log4j.appender.A1.layout.ConversionPattern=%d [%t] %-5p %c - %m%n

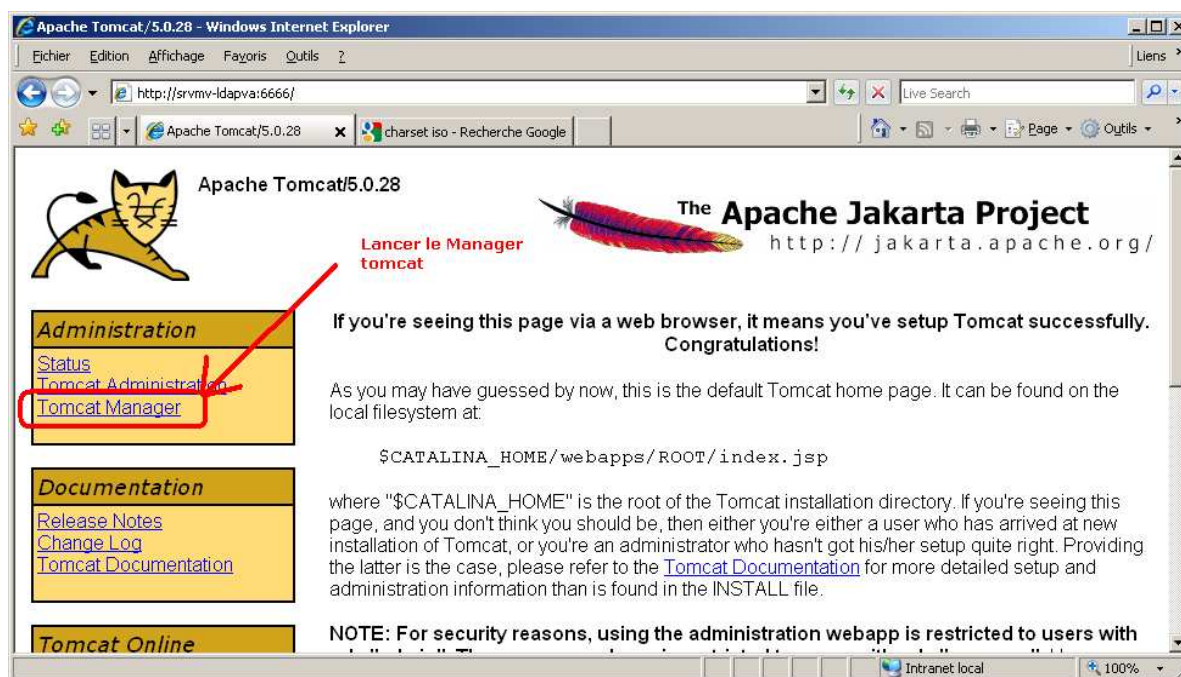
# configuration for a rolling "erreurs" log file
log4j.appender.R=org.apache.log4j.RollingFileAppender
log4j.appender.R.File=${catalina.home}/logs/errors.log
log4j.appender.R.MaxFileSize=10MB
log4j.appender.R.MaxBackupIndex=5
log4j.appender.R.layout=org.apache.log4j.PatternLayout
log4j.appender.R.layout.ConversionPattern=%d [%t] %-5p %c - %m%n
#
```

7) Suppression des anciens logs sous **/opt/tomcat/logs**

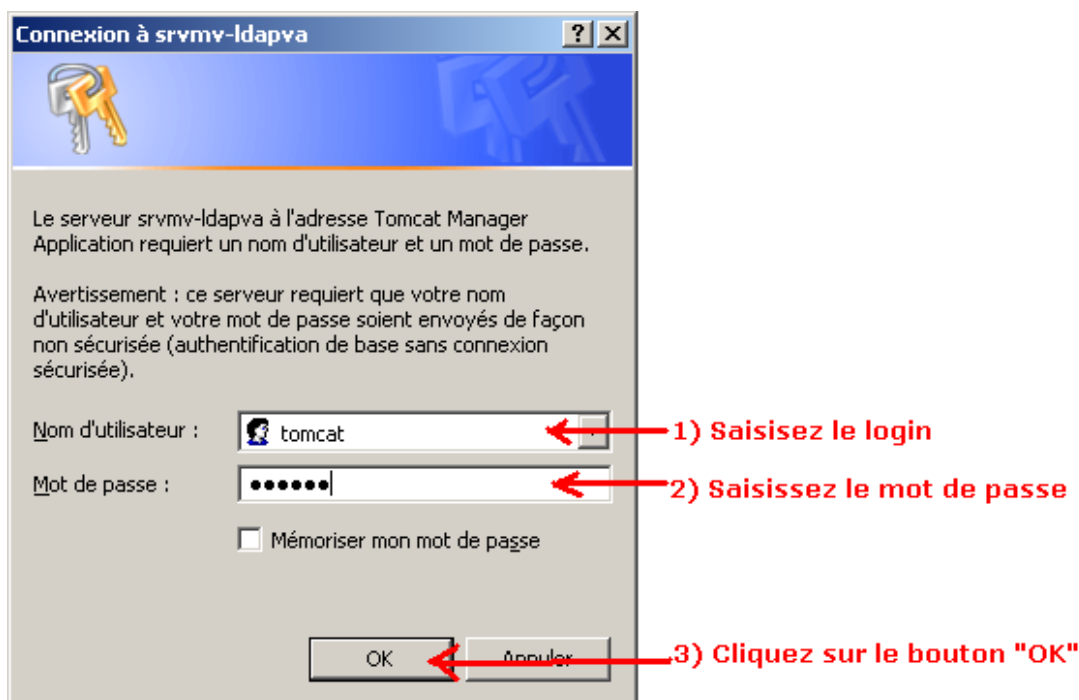
8) Relance du serveur tomcat (**/etc/init.d/tomcat start**)

4.10. Vérification de déploiement correct

Lancer un navigateur web et accéder à la console Manager de Tomcat via l'URL : **http://serveur:port**
Vous devez obtenir ceci :



Une fois que vous avez cliqué sur le lien [Tomcat Manager](#), vous obtenez une fenêtre d'authentification. Saisissez alors les logins (**tomcat** par défaut) et mot de passe (**tomcat** par défaut) d'administration puis cliquez sur le bouton OK :



Remontez éventuellement en tête de la page du Manager Tomcat et vérifiez que vous visualisez bien « **AIP_CG33** » dans la liste des applications et que celle-ci est bien en cours d'exécution :

Windows Internet Explorer

http://srvmv-ldapva:6666/manager/html/upload

The Apache Jakarta Project
http://jakarta.apache.org/

Application AIP_CG33 déployée

Gestionnaire d'applications WEB Tomcat

Message: OK - Application déployée pour le chemin de contexte /AIP_CG33. Application en cours d'exécution, Démarrer n'est pas sélectionnable.

Message de confirmation de déploiement réussi

Manager

List Applications HTML Manager Help Manager Help Etat du serveur

Applications

Chemin	Nom d'affichage	Fonctionnant	Sessions	Commands
/	Welcome to Tomcat	true	0	Démarrer Arrêter Recharger Undeploy
/AIP_CG33		true	0	Démarrer Arrêter Recharger Undeploy
/admin	Tomcat Administration Application	true	0	Démarrer Arrêter Recharger Undeploy
/balancer		true	0	Démarrer Arrêter Recharger Undeploy

Intranet local 100%

5. Phase 3 - Paramétrage de base obligatoire

5.1. Etape N°1 : Arrêtez tomcat

Sous linux, en tant que user **tomcat** :

- 1) Lancez la commande **/etc/init.d/tomcat stop**
- 2) Effacer tous les fichiers de log sous **\$(CATALINA_HOME)/logs** (pour « partir propre »)

5.2. Etape N°2 : Paramétrage minimal

Le logiciel A.I.P. v 1.0 s'appuie sur un fichier de paramétrage dans lequel sont définies les coordonnées de connexion à l'annuaire LDAP (on rappelle que ce dernier n'est pas forcément installé sur la même machine que le logiciel AIP v 1.0).

Il faut donc mettre à jour ces paramètres dans **{CATALINA_HOME}/webapps/AIP_CG33/conf/params.XML**

Pour ce faire, procédez comme suit :

- 1) Sous linux, éditez (par exemple avec vi ici) le fichier et mettez à jour les paramètres suivants avec les bonnes valeurs :

Paramètre	Exemple de valeur	Commentaire
Connexions à l'annuaire LDAP		
initialContextFactory	com.sun.jndi.Ldap.LdapCtxFactory	Contexte d'accès au serveur LDAP (mode JNDI)
providerURL	ldap://serveur :port	URL LDAP d'accès au serveur d'annuaire
securityAuthentication	simple	Type d'authentification sur le serveur le LDAP
baseRecherche	dc=cg33,dc=cg33	Racine de base de recherche dans le LDAP
securityPrincipal	cn=cg33,dc=cg33,dc=cg33	Utilisateur de consultation de l'annuaire LDAP
securityCredentials	*****	Mot de passe de l'utilisateur
topLevel	cg33	Libellé de la racine de haut niveau dans le LDAP
Charsets (les valeurs fournies ici sont normalement correctes pour une machine virtuelle Linux)		
ldapServerCharsetEncoding	UTF-8	Charset encoding du serveur LDAP
serveurCharsetEncoding	UTF-8	Charset encoding du serveur sur lequel tourne le Tomcat qui sert l'application AIP v 1.0
exportCharsetEncoding	ISO-8859-1	Charset encoding du fichier d'export de données au format XLS (ISO-8859-1 en principe)
mailCharsetEncoding	ISO-8859-1	Charset encoding du fichier joint dans le mail envoyé au support lors d'une demande de modification par un agent.
Répertoire des photos des Agents		
repPhotos	photos	Par défaut, la livraison contient un répertoire photos en local avec un exemple de fichier photo à l'intérieur. MAIS : il faut en fait que ce répertoire soit un montage sur le lecteur réseau qui stocke toutes les photos des agents, à savoir au CG33 \lsrvphoto\DRH\EDIGRAPH\CG33\Images
Paramétrage des envois de mail en cas de modification de fiche Agent		
mailMappingChampsModifies	photo,departmentNumber,commentaires,sn,givenName,ou,organizationalUnitName,title,manager,secretary,to:o.chator@cg33.fr,cc:o.chator@cg33.fr	A adapter selon votre chaîne de traitement des fiches agent modifiées. Les noms de champs sont ceux du LDAP, les destinataires (séparés par des virgules) derrière « to : », les copies (séparés par des virgules) derrière « cc : » et les différents blocs d'envois séparés par des « ; »

☞ Remarque :

Normalement, seuls les paramètres **providerURL** et **xxxxxxCharsetEncoding** devraient être à vérifier car :

- **providerURL** contient le numéro de serveur et de port permettant d'échanger avec l'annuaire LDAP. Par exemple, sur le serveur de VA du logiciel AIP au CG33, cette variable a pour valeur **ldap://serveur :0000**. Rapprochez vous des collègues des infrastructures ou du SPSI pour avoir les coordonnées exactes si nécessaires.
- **xxxxxxCharsetEncoding** contenant les définitions de codage de caractère (souvent **ISO-8859-1** pour ce qui va vers du monde Windows et **UTF-8** sous Linux)

5.3. Etape N°3 : Effacement manuel de fichiers

Supprimez manuellement les fichiers suivants :

- **\${CATALINA_HOME}/AIP_CG33/logs/ASupprimer.txt**
- **\${CATALINA_HOME}/AIP_CG33/temp/ASupprimer.txt**

5.4. Etape N°4 : Créer le lien vers le montage rése au

Au CG33, l'AIP s'appuie sur le référentiel des photos (fichiers jpg) géré par le logiciel EDIGRAPH. Il conviendra donc de réaliser un montage réseau du répertoire **photo** dans la distribution AIP (voir plus loin dans ce document) vers le répertoire [\\srvphoto\DRH\EDIGRAPH\CG33\Images](#) en mode **lecture seule**.

Le lien doit être mis à jour (si cela n'a pas déjà été fait) avec la variable **repPhotos** dans le fichier de configuration **params.xml**

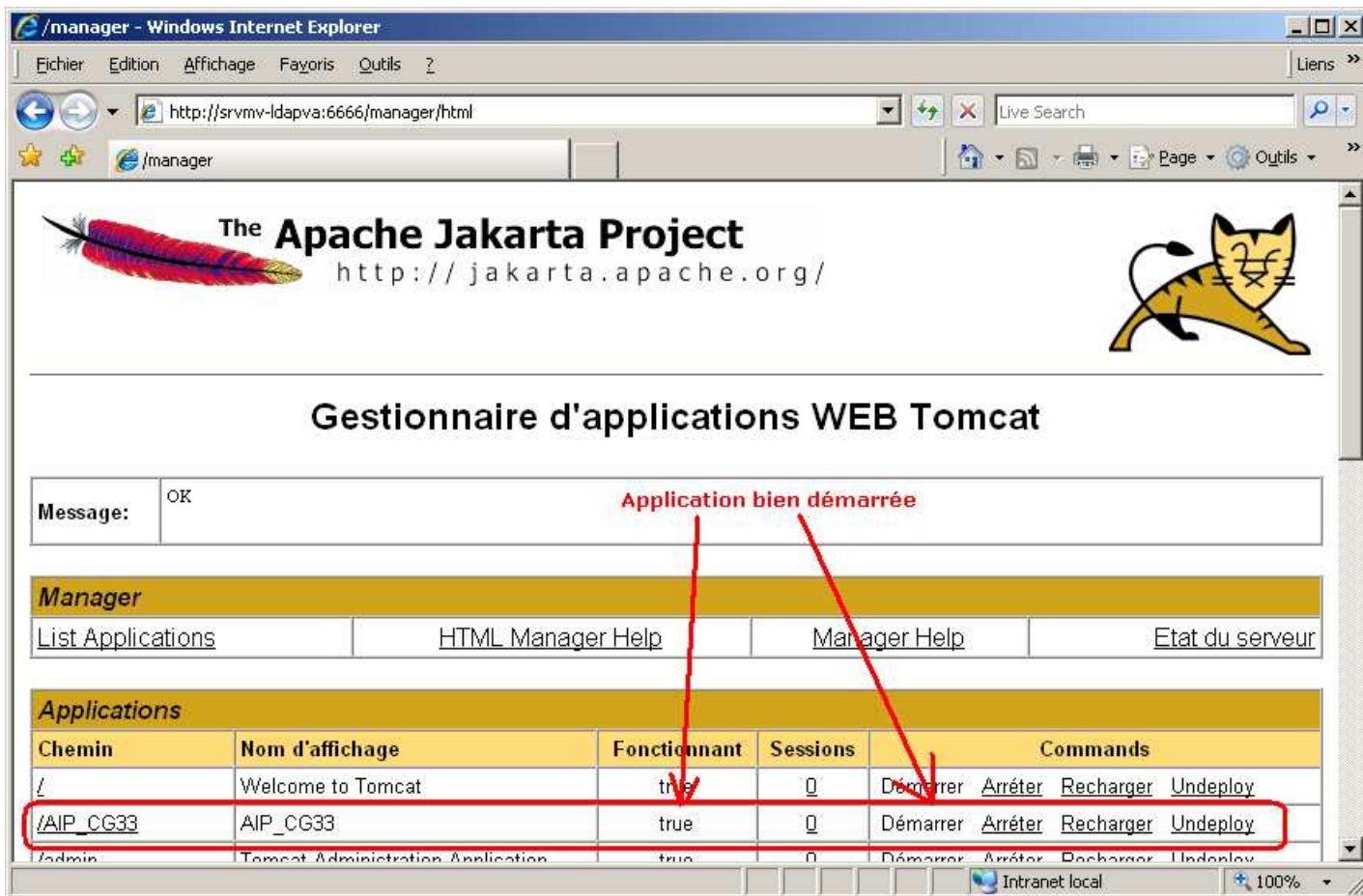
5.5. Etape N°5 : Relancez tomcat

Sous linux, lancez la commande **/etc/init.d/tomcat start**

6. Phase 4 - Vérifications post-installation

6.1. Via la console Manager de Tomcat

Suite aux opérations décrites au chapitre précédent, si vous lancez le Manager Tomcat, vous devez obtenir la fenêtre suivante dans laquelle l'application a bien démarré :



The screenshot shows the Tomcat Manager interface. At the top, it says 'The Apache Jakarta Project' with the URL 'http://jakarta.apache.org/'. Below this is the title 'Gestionnaire d'applications WEB Tomcat'. A message box indicates 'Message: OK' and 'Application bien démarrée'. Below the message box, there are links for 'List Applications', 'HTML Manager Help', 'Manager Help', and 'Etat du serveur'. The main section is titled 'Applications' and contains a table with the following data:

Chemin	Nom d'affichage	Fonctionnant	Sessions	Commands
/	Welcome to Tomcat	true	0	Démarrer Arrêter Recharger Undeploy
/AIP_CG33	AIP_CG33	true	0	Démarrer Arrêter Recharger Undeploy
/admin	Tomcat Administration Application	true	0	Démarrer Arrêter Recharger Undeploy

6.2. Via l'arborescence de fichier Tomcat

Vous devez constater, sous le répertoire `$(CATALINA_HOME)/webapps` qu'il existe bien le lien vers AIP_CG33 :

```
lrwxrwxrwx 1 tomcat tomcat 26 oct 30 11:05 AIP_CG33 -> /home/aip/webapps/AIP_CG33
```

6.3. Via les fichiers de log Tomcat

🔑 Exemple de contenu à la fin du catalina.out suite à une relance correcte (pas d'erreur) :

```
INFO: Installation d'une application pour le chemin de contexte /AIP_CG33 depuis l'URL  
file:/opt/tomcat/webapps/AIP_CG33  
25 févr. 2009 17:32:17 org.apache.coyote.http11.Http11Protocol start  
INFO: Démarrage de Coyote HTTP/1.1 sur http-6666  
25 févr. 2009 17:32:17 org.apache.jk.common.ChannelSocket init  
INFO: JK2: ajp13 listening on /0.0.0.0:8009  
25 févr. 2009 17:32:17 org.apache.jk.server.JkMain start  
INFO: Jk running ID=0 time=2/66 config=/opt/tomcat/conf/jk2.properties  
25 févr. 2009 17:32:17 org.apache.catalina.startup.Catalina start  
INFO: Server startup in 1653 ms
```

🔑 Exemple de contenu à la fin du localhost_log.yyyy-mm-dd.txt suite à une relance correcte (pas d'erreur) :

```
2009-02-25 17:32:16 StandardContext[/balancer]org.apache.webapp.balancer.BalancerFilter:  
init(): ruleChain: [org.apache.webapp.balancer.RuleChain:  
[org.apache.webapp.balancer.rules.URLStringMatchRule: Target string: News / Redirect URL:  
http://www.cnn.com], [org.apache.webapp.balancer.rules.RequestParameterRule: Target param  
name: paramName / Target param value: paramValue / Redirect URL: http://www.yahoo.com],  
[org.apache.webapp.balancer.rules.AcceptEverythingRule: Redirect URL:  
http://jakarta.apache.org]]  
2009-02-25 17:32:17 StandardContext[/servlets-examples]ContextListener:  
contextInitialized()  
2009-02-25 17:32:17 StandardContext[/servlets-examples]SessionListener:  
contextInitialized()  
2009-02-25 17:32:17 StandardContext[/jsp-examples]ContextListener: contextInitialized()  
2009-02-25 17:32:17 StandardContext[/jsp-examples]SessionListener: contextInitialized()  
2009-02-25 17:35:47 StandardContext[/manager]HTMLManager: init: Associated with Deployer  
'localhost'  
2009-02-25 17:35:47 StandardContext[/manager]HTMLManager: init: Global resources are  
available  
2009-02-25 17:35:47 StandardContext[/manager]HTMLManager: list: Listing contexts for  
virtual host 'localhost'
```

6.4. Via l'application AIP_CG33 elle-même

Via votre navigateur, enchaînez rapidement quelques écrans de l'application. URL : **http://serveur:port/AIP_CG33**
Le fait que cet écran apparaisse traduit que la connexion au LDAP est correcte et que des données sont bien échangées avec lui (la liste *DGA ou Direction ou Service* ne serait pas complétée en cas de problèmes)



http://aipva.cg33.fr:8080/AIP_CG33/AfficherFormulaireRecherche - Windows Internet Explorer

Fichier Edition Affichage Favoris Outils ?

http://aipva.cg33.fr:8080/AIP_CG33/AfficherFormulaireRecherche

Google

Favoris http--intranuaire-AIP_CG3... AIP BDP Deploy Manager DSI Espaces Co

http://aipva.cg33.fr:8080/AIP_CG33/AfficherFormulaireRecherche

Annuaire interne des Personnels

Gironde CONSEIL GENERAL

Rechercher

Nom commence par

Nom contient

Prénom contient

Téléphone (numéro de poste)

Libellé court de Service cg33

Libellé long de Service CONSEIL GENERAL DE LA GIRONDE

Mot clé

Organigramme et explorateur des services

Effacer Rechercher

Terminé Internet 95%

7. Phase 5 - Désinstallation

7.1. Désinstallation

7.1.1. Des pré-requis

Si les services sont actifs, passez les commandes :

```
/etc/init.d/tomcat stop
```

```
/etc/init.d/openldap stop
```

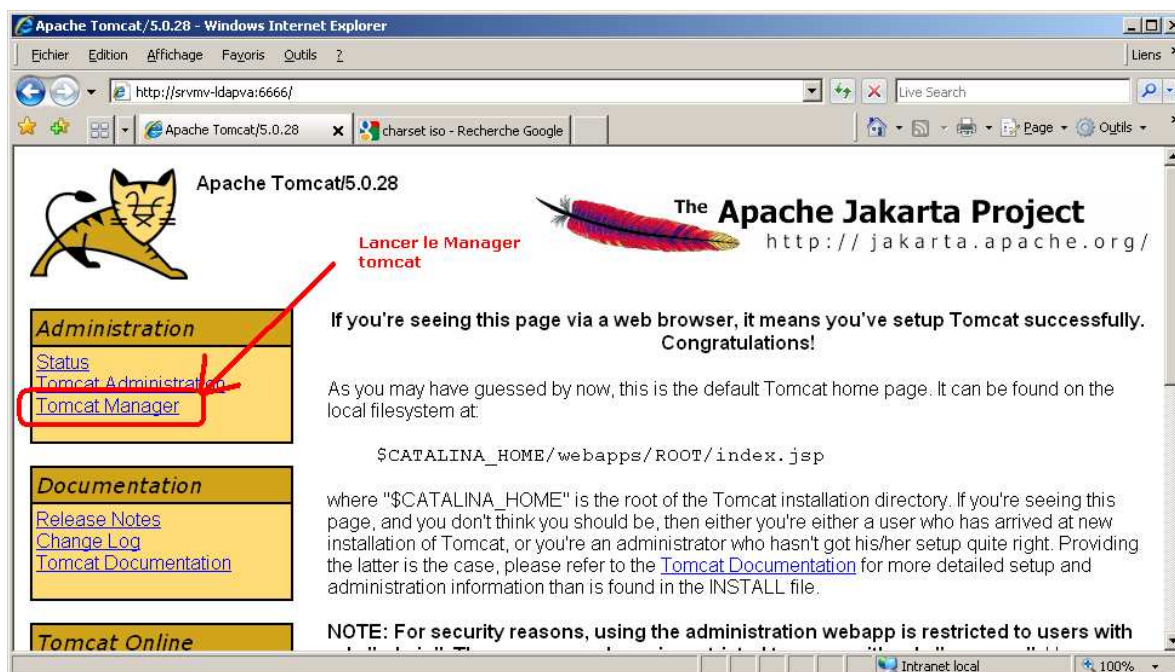
Il convient simplement de supprimer les répertoires suivants et tout ce qu'ils contiennent :

- `${CATALINA_HOME}`
- `${JDK_HOME}`
- `${LDAP_HOME}`
- `${BERKELEY_HOME}`

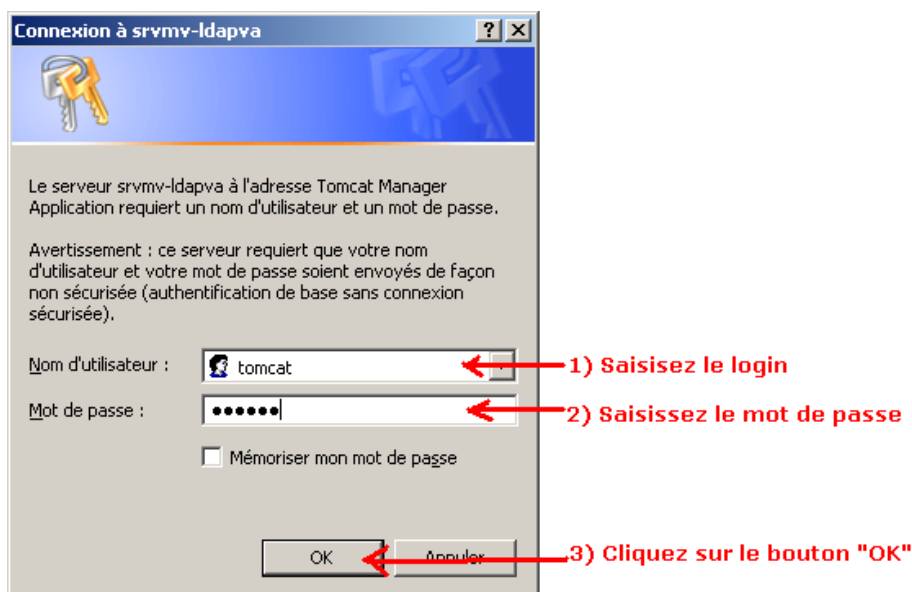
7.1.2. De l'AIP via la console d'administration Tomcat

Enchaînez les actions suivantes :

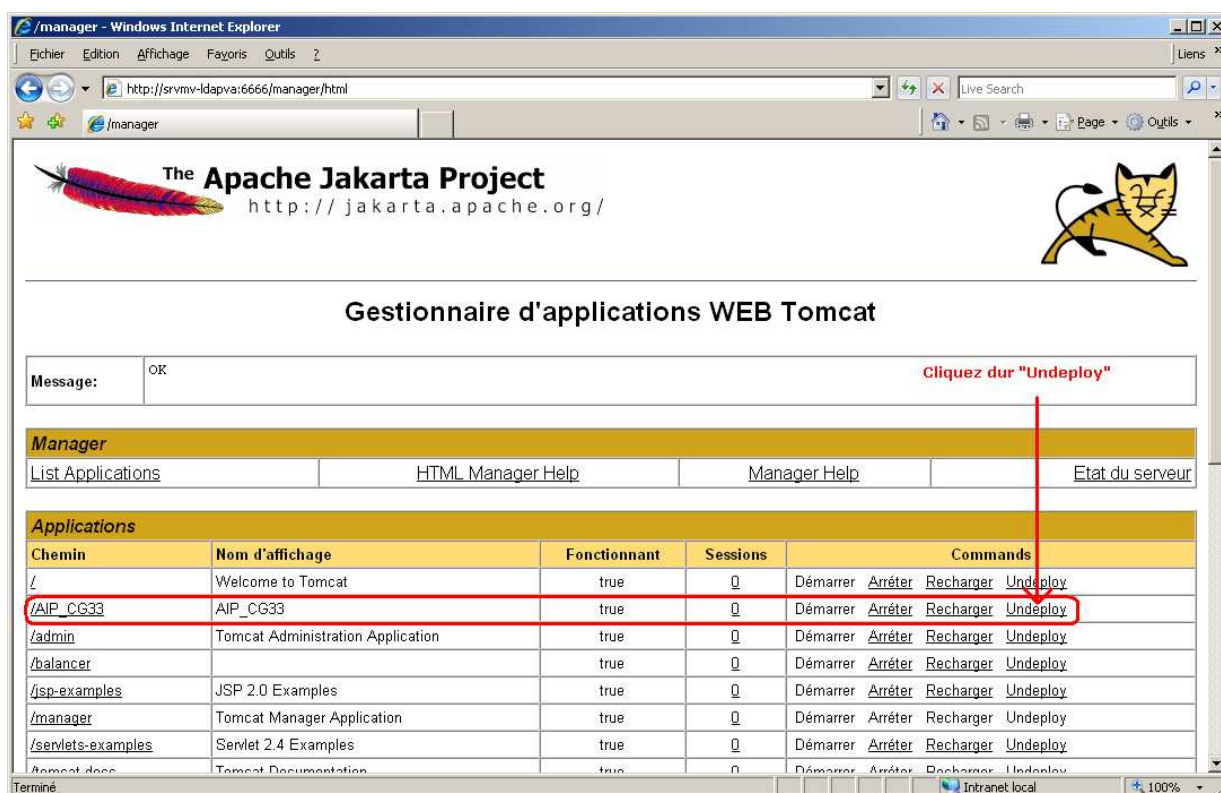
- 1) Accédez à la console Manager de Tomcat via l'URL : **`http://serveur:port/`**
Vous devez obtenir ceci :



- 2) Une fois que vous avez cliqué sur le lien **Tomcat Manager**, vous obtenez une fenêtre d'authentification. Saisissez alors les logins (**tomcat** par défaut) et mot de passe (**tomcat** par défaut) d'administration puis cliquez sur le bouton **OK** :



3) Sur la fenêtre qui apparaît, sélectionnez le lien « **Undeploy** » :



7.1.3. De l'AIP de manière manuelle

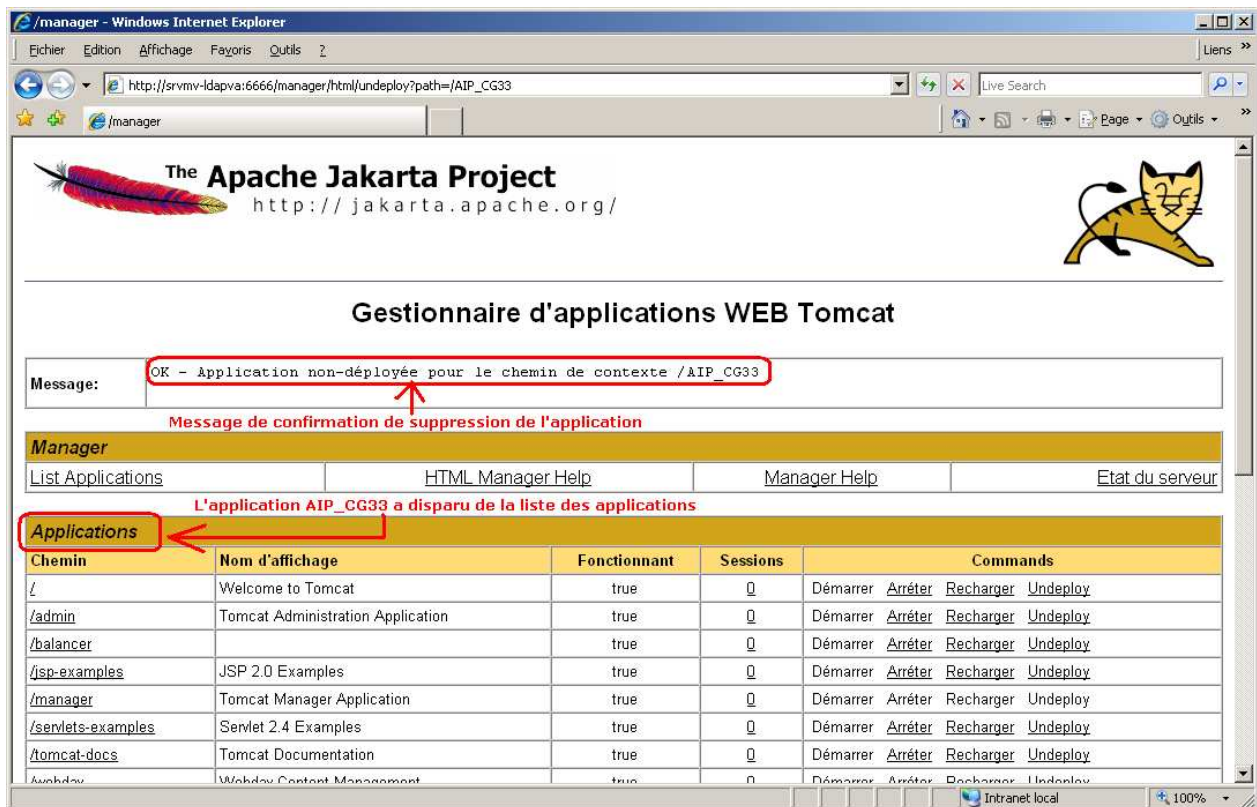
Pour effectuer une désinstallation manuelle, procédez comme suit :

- 1) Exécuter : **/etc/init.d/tomcat stop**
- 2) Exécuter : **rm -r /home/aip/webapps/AIP_CG33**
- 3) Exécuter : **rm /opt/tomcat/webapps/AIP_CG33**
- 4) Exécuter : **/etc/init.d/tomcat start**

8. Phase 6 – Vérifications post désinstallation

8.1. Via la Console Manager de Tomcat

En fin de désinstallation (cf. phase précédente), vous devez obtenir la fenêtre suivante. **AIP_CG33** a été supprimée de la liste des applications déployées et un message de confirmation a été fourni dans la console Manager :



8.2. Via les Fichiers de log Tomcat

☛ Exemple de contenu à la fin du catalina.out suite à une désinstallation correcte :

```
26 févr. 2009 09:04:18 org.apache.catalina.core.ContainerBase log
INFO: Retrait de l'application web pour le chemin de contexte /AIP_CG33
```

☛ Exemple de contenu à la fin du localhost_log.yyyy-mm-dd.txt suite à désinstallation correcte :

```
2009-02-26 09:04:18 StandardContext[/manager]HTMLManager: undeploy: Undeploying web
application at '/AIP_CG33'
2009-02-26 09:04:18 StandardContext[/manager]HTMLManager: Undeploying document base
/opt/tomcat/webapps/AIP_CG33
```

9. Annexe : Intervention sur les données de l'annuaire

9.1. Introduction

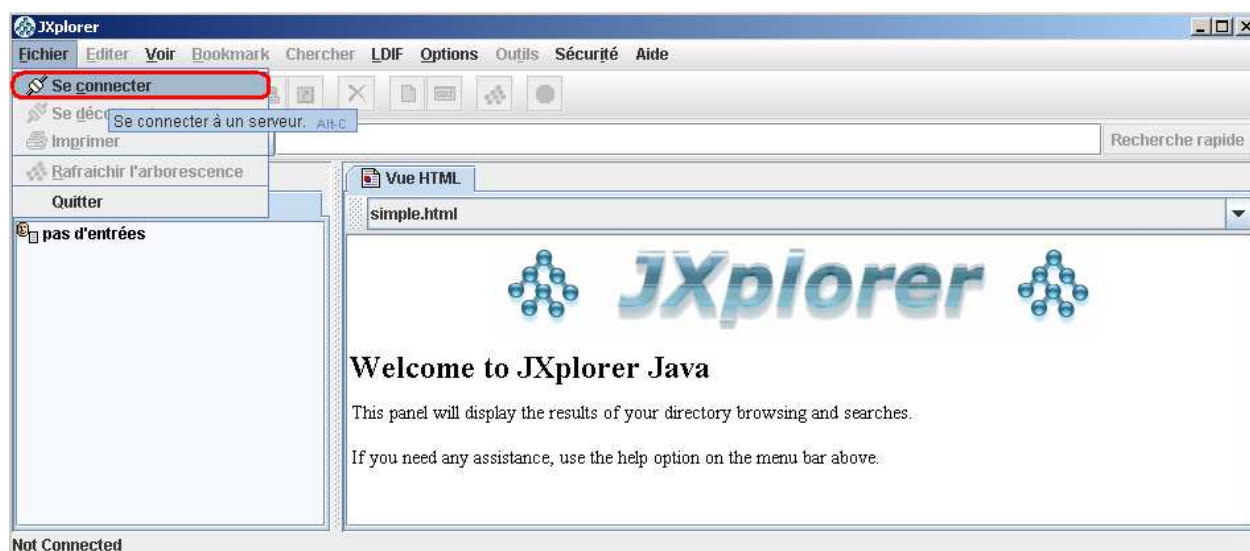
Ce chapitre **n'est à mettre en œuvre qu'en cas d'urgence élevée** où une modification en direct sur les données s'avère nécessaire. Il ne doit être utilisé dans aucun autre cas puisque l'alimentation en données est issue d'interface automatisées à partir des applications métier amont

9.2. L'outil JExplorer

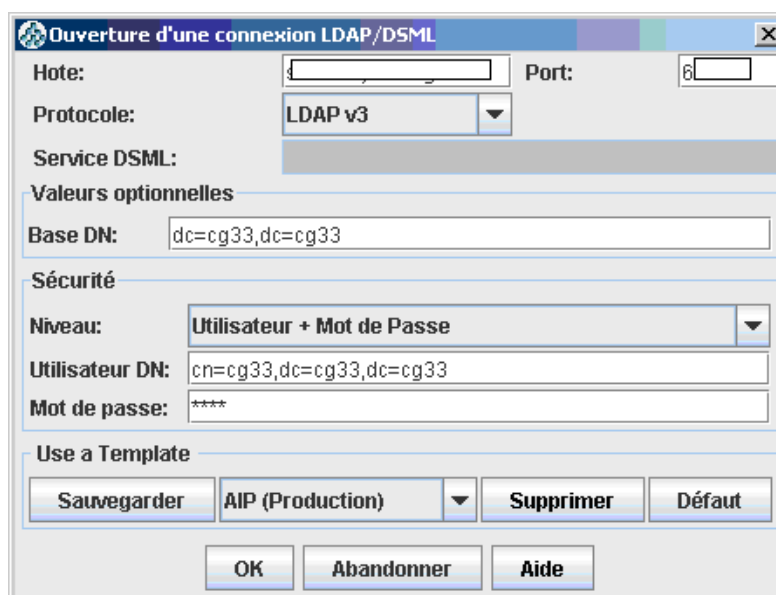
C'est un outil Open Source qui peut être déployé sur un client Linux ou Windows puisqu'il est entièrement écrit en langage Java. Il est directement téléchargeable sur le site : <http://jxplorer.org/>

9.3. Connexion au Ldap

Etape 1 : Sur l'interface graphique de l'outil, sélectionner **Fichier -> Se connecter** :

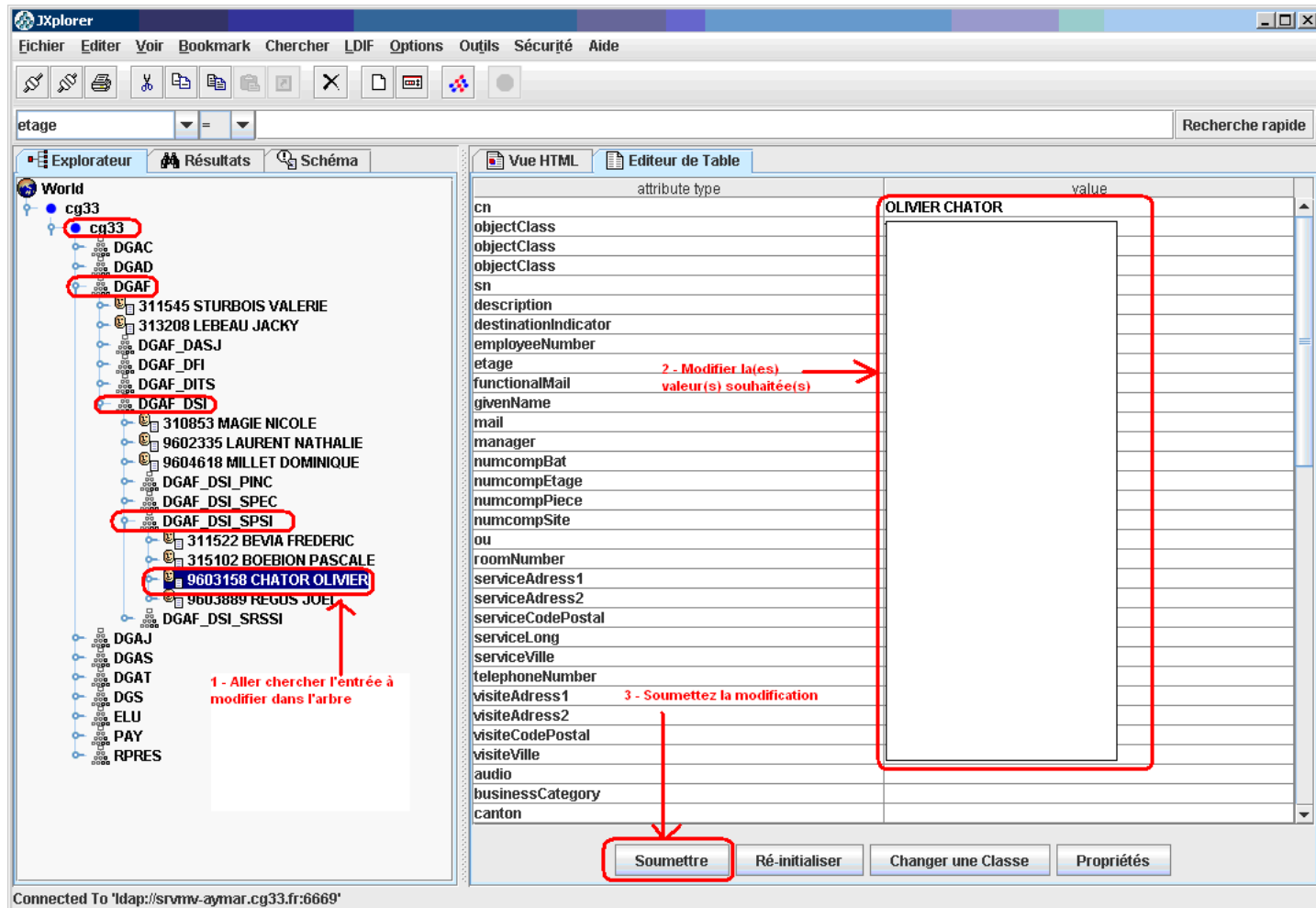


Etape 2 : Remplir les données de connexion à l'annuaire Open LDAP puis cliquer sur le bouton **OK** :



9.4. Modification de données

Utilisez l'arbre de gauche pour aller chercher l'entrée à modifier puis l'onglet « **Editeur de table** » sur celui de droite pour visualiser toutes les données de l'entrée choisie.

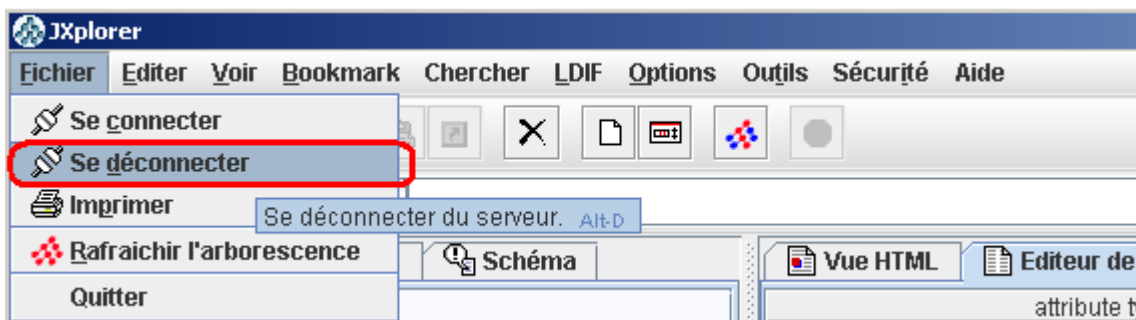


The screenshot shows the JXplorer application window. On the left, the 'Explorateur' pane displays a tree structure of LDAP entries. The entry '9603158 CHATOR OLIVIER' is selected and highlighted with a red box. A red arrow points to it with the text '1 - Aller chercher l'entrée à modifier dans l'arbre'. On the right, the 'Editeur de Table' pane shows the details of the selected entry. The 'cn' field is highlighted with a red box and contains the value 'OLIVIER CHATOR'. A red arrow points to this field with the text '2 - Modifier la(es) valeur(s) souhaitée(s)'. At the bottom of the 'Editeur de Table' pane, the 'Soumettre' button is highlighted with a red box and a red arrow points to it with the text '3 - Soumettez la modification'. The status bar at the bottom indicates 'Connected To 'ldap://srvmv-aymar.cg33.fr:6669''.

Le fait de cliquer sur le bouton « Soumettre » implique une modification immédiate des données qui est répercutée aussitôt sur l'IHM de consultation de l'annuaire.

9.5. Déconnexion

Sélectionnez **Fichier -> Se déconnecter**



The screenshot shows the JXplorer application window with the 'Fichier' menu open. The 'Se déconnecter' option is highlighted with a red box. The status bar at the bottom indicates 'Se déconnecter du serveur. Alt-D'.